

## REMARKS

### Pending claims

Assuming entry of this amendment, claims 1-45 are still pending or allowed, of which claims 1, 4, 13, 25, 26, 27, 28, 30, 34, 39, 44 and 45 are independent. The applicants are not requesting any amendments to the claims that were previously or originally filed.

### Claim Rejections & Rejections Under 35 U.S.C. 103(a)

The Examiner rejected claims 1-3, 24, 28 and 29 under 35 U.S.C. 103(a) as being unpatentable over Bonola (U.S. Patent 5,913,058 – "*Bonola*") in view of Sato et al. (U.S. Patent 4,835,677 – "*Sato*"). Of these, claims 1 and 28 are independent.

In particular, tracking claim 1, the Examiner wrote that *Bonola* teaches all the features of claim 1 (and 28) except the feature of the kernel substantially displacing the COS, including scheduling execution of the COS on the hardware processor. The Examiner asserts, however:

Sato teaches the kernel substantially displacing the COS, via the kernel, including scheduling execution of the COS on the hardware processor (the first operating system being provided with access to said first control registers by the supervisory operating system; the supervisory operating system writing a first value in the register indication bit of said extended control register when the first operating system run, col 3, ln. 57-61 / ln 65-67 to col 4, ln 1-2 / col 2, ln 34-40 / ln 49- ln 50-51 / ln 55-58 and abstract ln 8-13).

It would have been obvious to one of the ordinary skill in the art at the time the invention was made to combine the teaching of Bonola and Sato because Sato's the kernel substantially displacing the COS, via the kernel, including scheduling execution of the COS on the hardware processor would reduce the overall overhead time of the system when the hardware can be accessed directly by the operating system.

It is unclear which of *Sato's* operating systems – the control program (CP) 22 or one of the operating systems (OS1, OS2) 27, 28 – the Examiner believes is analogous to the applicants' first operating system COS and kernel, respectively, but regardless of which is "matched" with which, *Sato* still lacks an essential feature of the applicants' invention as recited in the independent claims 1 and 28: In *Sato*, no software module

that initializes the computer (presumably, the control program CP, if any) is ever displaced by any other later-loaded software module such that the later-loaded module also schedules execution of the initializing module on any hardware processor.

Rather, in *Sato*, the control program CP at all times not only controls the extended control register (col. 2, lines 32-34), but also "supervises operating systems (OS1) 27 and (OS2) 28" (col. 2, lines 34-35). Note that without a supervisory software layer always present, there would be no way to resolve possible contention for access by OS1 and OS2 to the shared control registers CR0, CR1, CR2, CR3, CR8, CR9, CR10, CR11, CR14 and CR15.

*Sato's* Figure 1 shows that even when OS1 has "direct" access to the control registers, then this is by way of the control program CP rather than by bypassing the CP altogether. There is no indication in *Sato* that there would ever be a complete context switch away from the control program; in fact, if this were the case then the control program would never know when OS1 had stopped running and OS2 had started to run. This is also made clear in *Sato* itself (emphasis added):

the first operating system [OS1 is] provided with access to [the] first control registers **by the supervisory operating system** [CP] (col. 3, lines 57-60);  
the second operating system [OS2 is] provided with access to [the] second control registers **by the supervisory operating system** [CP]; (col. 3, lines 61-64)

That *Sato's* OS1's control register load/store instructions may be executed directly by the hardware process means only that the control program CP need not invoke additional code to transfer access to extended registers CR6, CR7 as a result of a CP interrupt mechanism. In short, *Sato's* CP is *always* at a supervisory level with respect to OS1 and OS2 and is never displaced from this function by either operating system OS1 or OS2.

Assuming that one virtual machine's OS (OS1, OS2) has, even temporarily, exclusive access to the control registers is irrelevant – this is what happens whenever any supervisory operating system schedules different applications on a hardware

processor: At all times, *Sato's* control program CP is still the only software module with a function resembling scheduling (which, in *Sato's* case, also involves setting the control bit 24 to "1" or "0" depending on whether OS1 or OS2 is currently running). Neither OS1 nor OS2 ever takes supervisory control of the control program CP. Thus, *Sato* does *not* teach any form of kernel "substantially displacing [a] COS from the system level and itself running at the system level; and handling requests for system resources via the kernel, including scheduling execution of the COS on the hardware processor(s)."

It would also not be obvious to combine the teachings of *Bonola* and *Sato*, because, among other reasons, *Bonola* would have no need for *Sato's* interrupt mechanism – his real-time kernel executes on a **dedicated** I/O processor (col. 5, lines 9-12). As such, *Bonola's* kernel has exclusive access to the registers of the I/O processor, with no need to share, and thus no need for any mechanism that coordinates access to some other set of simulated registers located in memory as in *Sato*.

As explained in the response to the previous Office action, once *Bonola's* kernel 30 is loaded, the host operating system remains in control of access to all other system resources, in particular, to the remaining processors 12a-12c (see, for example, col. 10, lines 3-8). *Bonola's* host operating system then continues to function as a conventional operating system in that it still controls access to all system resources except the I/O processor 12d and its associated device 24. To even have a need for *Sato* would presuppose that *Bonola's* various operating systems all share the same set of processors, rather than having a separate kernel running on a dedicated processor. In other words, incorporating *Sato* would presuppose a configuration that is opposite to the entire stated purpose of *Bonola*.

Because neither *Bonola* nor *Sato* discloses 1) a first operating system that initializes the computer, 2) then loads and starts execution of a kernel, 3) but that is then functionally displaced from the system level by the kernel, 4) which then also schedules execution of the first operating system on the hardware processor(s) – all features recited in claims 1 and 28 – no combination of these references would either.

The hypothetical combination therefore cannot render the applicants' invention obvious because even the combination would still lack key claimed features. Consequently, the invention as claimed in claims 1 and 28 should be patentable since it is both novel and non-obvious, and provides clear technical advantages that are discussed at length in the specification.

The Examiner also specifically rejected the claims that depend from claims 1 and 28 as being obvious in view of *Bonola*, *Sato*, either alone or in further in combination with either "admitted prior art" or Bugnion et al. (US 6,017,938). These dependent claims by definition incorporate all of the limitations of their respective independent base claims. Since the base claims 1 and 28 should be allowable, so should their respective dependent claims.

#### **Request to Withdraw Finality of Rejection**

As explained above, the claims should be allowed without further amendment. Nonetheless, if the Examiner still feels there is a substantial issue preventing allowance, the applicants respectfully request not only that the finality of the 14 December 2004 Office action be withdrawn, but also that they be given another Office action on the merits without the need to file a continuation application.

In this second Office action, the Examiner cited *Sato* (as a secondary reference in combination with *Bonola*) for the first time, even though *Sato* was available to the Examiner in preparing the first Office action. Wanting to avoid delay in prosecution, the applicants responded promptly to the first Office action and explained the patentability of the invention over *Bonola* and *Solomon*.

Now it appears that the Examiner did not consider *Solomon* to be the best secondary reference after all (M.P.E.P. 904.03: "The best reference should always be the one used."), since he now rejects independent claims 1 and 28 based in part on a different reference – *Sato* – that also was available in preparing the first Office action.

The applicants have at every point in the prosecution of this application done everything possible to facilitate a "speedy and just determination of the issues involved

in the examination" (M.P.E.P. 904.03) and at the time of filing, they presented "claims varying from the broadest to which they believe they are entitled to the most detailed that they would be willing to accept" (also M.P.E.P. 904.03). Thus, the applicants did everything the regulations encourage to facilitate and expedite prosecution, but they are faced with a piecemeal presentation of references and rejections (M.P.E.P. 707.07(g): "Piecemeal examination should be avoided as much as possible"). Moreover, nothing in the record indicates that any amendment by the applicants necessitated the new ground of rejection by the Examiner (M.P.E.P. 706.07(a)). If the Examiner does not agree that the present claims are allowable, then the applicants therefore respectfully assert that an opportunity to respond to another Office action on the merits would greatly further the Office's stated goal of a speedy and just determination of the issues involved in the examination of this application.

#### **Allowed Claims**

The Examiner has allowed claims 4-11, 13-20, 25-27, 30-32, 34-40, 44 and 45.

#### **Conclusion**

All of the original and previously presented claims have now either already been allowed, or should be allowable, as explained above. The applicants therefore respectfully request allowance even of claims 1-3, 24, 28 and 29, without having to pay to prepare and file a continuation application.

Date: 14 January 2005

VMware, Inc.  
3145 Porter Drive  
Palo Alto, CA 94304  
Phone & fax: (360) 793-6687

Respectfully submitted,



Jeffrey Pearce  
Reg. No. 34,729  
Attorney for the Applicants